

4. (Amended) The method according to claim 3, wherein the step of computing said two remaining vertices comprises:

A1
computing a vector given by a cross product between a normalized vector along said edge casting shadows and a normal onto a polygon surface associated to said edge casting shadows and setting the remaining two vertices of said additional polygon to points at a predetermined distance from said two vertices in a direction given by said computed vector.

5. (Amended) The method according to claim 4, wherein
the position of the remaining two vertices is further modified by computing interpolated vectors by respectively interpolating said computed vector with another vector correspondingly computed for an adjacent edge casting shadows and
the remaining two vertices of said additional polygon are respectively set to points at a predetermined distance from said two vertices in a direction given by said interpolated vector.

8. (Amended) The method according to claim 7, wherein the step of applying said cubemap while performing a stencil test operation comprises:

A2
performing a per-pixel test against values stored in the stencil buffer and preventing rendering a fragment of said scene at the pixel of interest if the value stored in the stencil buffer at the pixel of interest is different from the nil value;

accessing said cubemap via vectors given by 3D texture coordinates, where the greatest magnitude component is used to select a face of said cubemap and the other two components are used to select a texel from said face; and

performing a texture coordinate generation for specifying a texture coordinate and selecting a pixel of an appropriate cubemap face.

14. (Amended) The computer program product according to claim 13, wherein the program code means for computing said two remaining vertices comprises:

A3
cm.t

A3
Cmell

program code means for computing a vector given by a cross product between a normalized vector along said edge casting shadows and a normal onto a polygon surface associated to said edge casting shadows and setting the remaining two vertices of said additional polygon to points at a predetermined distance from said two vertices in a direction given by said computed vector.

15. (Amended) The computer program product according to claim 14, further comprising:

program code means for modifying the position of the remaining two vertices by computing interpolated vectors by respectively interpolating said computed vector with another vector correspondingly computed for an adjacent edge casting shadows and

program code means for setting the remaining two vertices of said additional polygon respectively to points at a predetermined distance from said two vertices in a direction given by said interpolated vector.

A4

18. (Amended) The computer program product according to claim 17, wherein a program code means for applying said cubemap while performing a stencil test operation comprises:

program code means for performing a per-pixel test against values stored in the stencil buffer and preventing rendering a fragment of said scene at the pixel of interest if the value stored in the stencil buffer at the pixel of interest is different from the nil value;

program code means for accessing said cubemap via vectors given by 3D texture coordinates, where the greatest magnitude component is used to select a face of said cubemap and the other two components are used to select a texel from said face; and

program code means for performing a texture coordinate generation for specifying a texture coordinate and selecting a pixel of an appropriate cubemap face.